

Claims

1. A tracking system (1) comprising means for receiving course player update data, and for writing the update data to a learning management database, characterised in that,
- the system (1) comprises database interface means for communication (24) with a plurality of learning management databases (10), and
- the system comprises a tracking engine (20) comprising means for managing uni-directional communication for asynchronous course player data updates, and for managing bi-directional communication for synchronous course player data updates and responses.
2. A system as claimed in claim 1, wherein the system further comprises a plurality of translation modules (22), each comprising means for translating from a player language to a common engine language, and vice-versa.
3. A system as claimed in claim 2, wherein each translation module is an object instantiated at start-up.
4. A system as claimed in claim 2, wherein the system (1) further comprises a common interface (21) comprising means for interfacing with all players (4), and for polling by routing a received message to all translation modules (22); wherein each translation module (22) comprises means for parsing received messages and, if it can translate the message, indicating as such; and wherein the common interface (21) comprises means for activating a translation module (22) which responds positively.

5. A system as claimed in claim 4, wherein the common interface (21) comprises means for polling the translation modules (22) according to a pre-set file.
6. A system as claimed in claim 5, wherein the common interface comprises means for receiving translated messages from the translation modules.
7. A system as claimed in claim 1, wherein the engine (20) comprises means for managing a queue (31), and for establishing threads for input and output to the queue.
8. A system as claimed in claim 7, wherein the engine (20) comprises means for assuming that a data update message is synchronous unless the message indicates otherwise.
9. A system as claimed in claim 1, wherein the database interface means (24) comprises a schema access object (SAO) associated with each learning management database (10).
10. A system as claimed in claim 9, wherein all SAOs (24) have the same exposed interface to the tracking engine (20).
11. A system as claimed in claim 9, wherein the system (1) comprises means for both pre-setting and for subsequently modifying associations between players and SAOs.
12. A system as claimed in any of claim 9, wherein the system (1) comprises pooling means, comprising means for creating a number of instances of each SAO and for reusing the instances.

13. A system as claimed in claim 12, wherein the pooling means comprises a manager comprising means for managing a pool of SAOs, for determining a free SAO if one exists, for putting a requesting thread in a sleep state if an instance is temporarily unavailable, and for instructing a new set of SAO instances to be created.
14. A system as claimed in claim 13, wherein the system (1) comprises means for determining during initialisation a connection string to be passed to an SAO instance to indicate the database (10) to be opened.
15. A system as claimed in claim 13, wherein the manager comprises means for re-initialising a pooled SAO instance if it operates incorrectly.
16. A system as claimed in claim 6, wherein the tracking engine (20) comprises means for maintaining an input thread (30) between the common interface (21) and the queue (31).
17. A system as claimed in any of claim 7, wherein the tracking engine (20) comprises means for maintaining a plurality of database-side threads (32) for routing messages from the queue (31) to the database interface means (24).
18. A system as claimed in claim 17, wherein each database-side thread (32) comprises means for waiting for a response from the database interface means (24), and for directly routing received responses to a relevant translation module (22) for translation and for receiving translated responses back from the translation modules.
19. A system as claimed in claim 18, wherein each database-side thread (32) comprises means for directly routing a translated response to an originating player.

20. A system as claimed in claim 17, wherein the engine (20) comprises means for maintaining each database-side thread (32) in an active state or a sleep state.
- 5 21. A system as claimed in claim 19, wherein the engine (20) comprises means for switching a database-side thread (32) to an active state in response to the input thread request.
- 10 22. A system as claimed in claim 1, wherein the engine (20) comprises means for writing the contents of the queue to a log file when shutting down unexpectedly, and for automatically searching for a log file upon start-up.
- 15 23. A system as claimed in claim 4, wherein the common interface (21) comprises a time-out function comprising means for terminating a player connection upon expiry of a pre-set time period.
- 20 24. A system as claimed in claim 1, wherein the database interface means comprises a time-out function comprising means for terminating a learning management database (10) connection upon expiry of a pre-set time period.
- 25 25. A tracking system (1) comprising means for receiving course player update data, and for writing the update data to a learning management database, characterised in that,
- the system (1) comprises database interface means for communication (24) with a plurality of learning management databases (10),
- the system comprises a tracking engine (20) comprising means for managing uni-directional communication for asynchronous course player data updates,

and for managing bi-directional communication for synchronous course player data updates and responses;

wherein the system further comprises a plurality of translation modules (22),
5 each comprising means for translating from a player language to a common engine language, and vice-versa;

wherein each translation module is an object instantiated at start-up; and

10 wherein the system (1) further comprises a common interface (21) comprising means for interfacing with all players (4), and for polling by routing a received message to all translation modules (22); wherein each translation module (22) comprises means for parsing received messages and, if it can translate the message, indicating as such; and wherein the common interface (21) comprises
15 means for activating a translation module (22) which responds positively.

26. A computer program product comprising software code for completing a system as claimed in any preceding claim when executing on a digital computer.

20